Microservices in Java
OSGi ist doch die Lösung

JavaLand 2017

28.03.2017

Microservices - Die Definition

MARTINFOWLER.COM

Intro Videos Design Agile Refactoring About Me All Sections ThoughtWorks

Microservices

a definition of this new architectural term

The term "Microservice Architecture" has sprung up over the last few years to describe a particular way of designing software applications as suites of independently deployable services. While there is no precise definition of this architectural style, there are certain common characteristics around organization around business capability, automated deployment, intelligence in the endpoints, and decentralized control of languages and data.

25 March 2014

James Lewis

James Lewis is a Principal Consultant at ThoughtWorks and member of the Technology Advisory Board. James' interest in building applications out of small collaborating services stems from a background in integrating enterprise systems at scale. He's built a number of systems using microservices and has been an active participant in the growing community for a couple of years.

Martin Fowler

Martin Fowler is an author, speaker, and general loud-mouth on software

Contents

https://martinfowler.com/articles/microservices.html

Microservices - Die Definition

- Jede Komponente läuft in einem eigenem Prozess
- Leichtgewichtige Kommunikation (meist Rest)
- Aufteilung in fachliche Komponenten
- Unabhängiges Deployment
- Unterschiedliche Verfahren zur Datenspeicherung
- Implementierung in unterschiedlichen Sprachen

- Microservice -> Modul/Container

Microservices - Die Definition

Pros

- Unabhängige Arbeit von Entwicklungsteams
- Konzentration auf eine Fachlichkeit
- Kleine abgeschlossene Komponenten

Cons

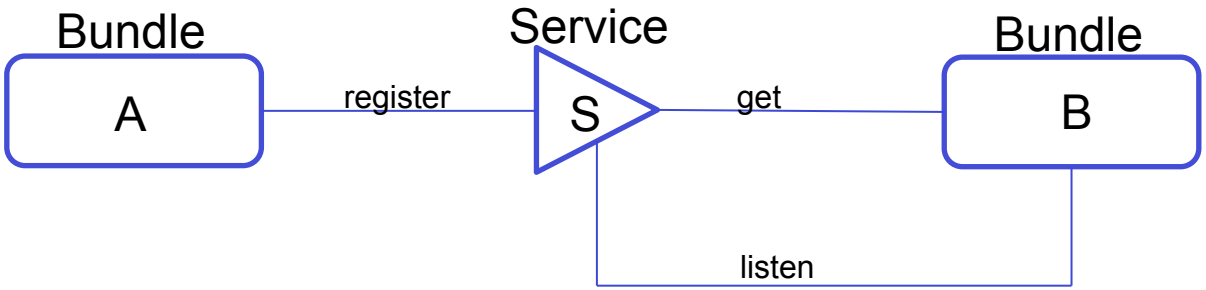
- Erhöhter Aufwand in
  - Betrieb
  - Deployment
  - Monitoring
- Kommunikationspfade können komplex sein
- Service-Dynamik

OSGi - Service-Platform

- Dynamisches Modulsystem für Java (JSR-291)
- Spezifizierung durch die OSGi-Alliance
- Aktuell Version R6
- R7 als Draft verfügbar

OSGi - Komponenten

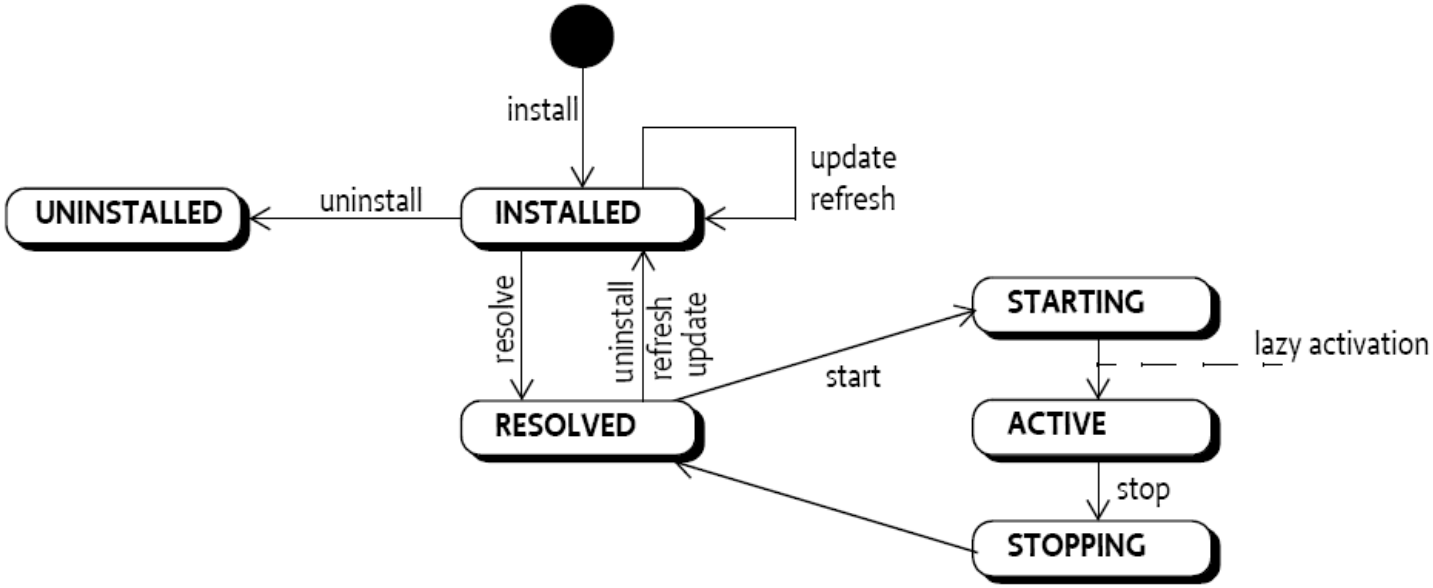
- Komponenten -> Bundles
- Bundles -> jar-Dateien mit erweiterten MANIFEST.MF
- Bundles können Services exportieren



OSGi - Bundle Manifest

```
Manifest-Version: 1.0
Bnd-LastModified: 1489929018728
Bundle-ManifestVersion: 2
Bundle-Name: rest-client
Bundle-SymbolicName: rest-client
Bundle-Version: 1.0.0.201703191310
Created-By: 1.8.0_121 (Oracle Corporation)
Private-Package: de.ndesign.javaland.cxf.rest.client
Import-Package: de.ndesign.javaland.cxf.rest.api;version="[1.0,2)",
 org.slf4j;version="[1.7,2)"
Provide-Capability: osgi.service;objectClass:List<String>="de.ndesign.javaland.cxf.rest.client.ShoppingListCommand"
Require-Capability: osgi.service;filter:="(objectClass=de.ndesign.javaland.cxf.rest.api.ShoppingListResource)";
effective:=active;cardinality:=multiple,osgi.ee;filter:="(&(osgi.ee=JavaSE)(version=1.8))"
Service-Component: OSGI-INF/de.ndesign.javaland.cxf.rest.client.ShoppingListCommand.xml
Tool: Bnd-3.3.0.201609221906
```

OSGi - Bundle Lifecycle



OSGi – Declarative Services

- Dependency Injection Framework
- Service-Binding
- Kompontenbeschreibung im XML-Format
- XML-Datei kann mit Hilfe von Annotationen automatisch erstellt werden.

OSGi – Remote Services/Remote Service Admin

- Veröffentlicht lokale Services im Netzwerk
- Ermittelt Endpunkte in Netzwerk
- Registrierung von entfernten Endpunkten als Service
- Keine Vorgaben bzgl. der zu verwendenden Protokolle

- Definiert ein Endpunkt-Beschreibungs-Format (XML)

OSGi - Aries RSA / CXF

- Implementierung Remote Admin Service
- Rest / SOAP – Schnittstelle
- Unterstüzung von declarative Services

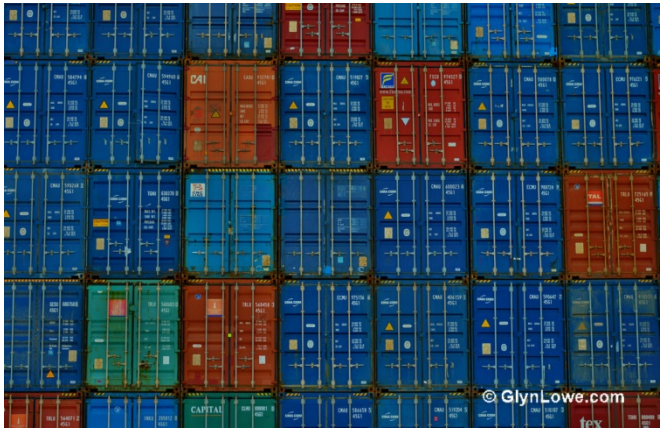
- Zookeeper Implementierung (http://zookeeper.apache.org/)

Demo (OSGi + Rest)

- Tool-Chain:
  - Eclipse
  - BND-Tools
  - gradle

Microservices + OSGi

- Jede Komponente läuft in einem eigenem Prozess ✔
- Leichtgewichtige Kommunikation (meist Rest) ✔
- Aufteilung in fachliche Komponenten ✔
- Unabhängiges Deployment ✔
- Unterschiedliche Verfahren zur Datenspeicherung ✔
- Implementierung in unterschiedlichen Sprachen ✔



Links

- Sourcen
  - https://github.com/uschulz/javaland2017
- Apache CXF
  - http://cxf.apache.org/distributed-osgi.html
- Apache Aries
  - http://aries.apache.org/modules/rsa.html
- Apache Zookeeper
  - http://zookeeper.apache.org/